Neil Sweigard
nas5183@psu.edu
STAT 580, Section 001 Summer 2022

Project 2 – Analysis Plan for House Price Predictions

For Project 2, I plan to feature engineer the original dataset to potentially enable additional analyses and stronger prediction for the client's original data. The first additional variable will be a `Neighborhood` categorical variable that will label which neighborhood's dataset a given observation is sourced from. A city's real estate prices can vary between neighborhoods, and this variable would be our only way to geographically distinguish homes. I will also split the `Exterior` column into `Exterior1st`, `ExteriorQual`, and `ExteriorCond` because there are several variables delimited within the original field. Similarly, I will split `LotInfo` into `LotConfig`, `LotShape`, `LotArea`, and `LotFrontage`. I will remove full row duplicates and an anomalous record with a `YrSold` of 2001. For missing values, I will impute "NA" for columns containing text and 0 for numeric columns. Because we do not have the `SalePrice` column to evaluate for the datasets labeled "test", I will perform a 75%/25% split of the non-test data into training and validation datasets that we can consistently evaluate each predictive model on. The Lasso, Ridge Regression, and Elastic Net methods require standardized quantitative predictor variables so I will also create a standardized version of the training and validation data. For other models, I will create dummy variables for the categorical variables.

The first family of models I will evaluate is a set of Multiple Linear Regressions. I will apply both Forward and Backward Stepwise selection methods then collect the selected variables that score best for each Mallow's Cp, BIC, and Adjusted R^2. With two selection methods multiplied by three selection criteria I will result in six MLR models to compare. I will then a fit models for Ridge Regression, Lasso, and Elastic Net. Each will require that I find distinct optimized tuning parameters. I will finally evaluate the tree-based family of models including Random Forest and XGBoost. Random Forest and XGBoost will require their own turning parameters that I will optimize with cross validation error.

To provide a fair comparison, each model will predict the `SalePrice` using the reserved validation data. Then, I will calculate the Mean Squared Error for each against the true `SalePrice` from the validation data as a simulation of performance against data that the model has not yet encountered before. I will ultimately select the model with the lowest MSE value to predict the `SalePrice` of the test datasets. Finally, I will output a csv file containing two columns for the `uniqueID` and the predicted `SalePrice` from my strongest performing model.